

## CLAIMS

Claim 1. A scalable fast block-matching motion estimation method for analyzing a plurality of video frames comprising:

employing search pattern scalability (that is, at least one block-matching fast motion estimation algorithm profile selected) and/or search distance computing scalability (that is, through sub-sampled pixel positions in computing macroblock distance) in order to adapt to application requirements and system's processing modes;

employing different search patterns designed for non-interlaced (or progressive-scanned) video;

employing different search patterns designed for interlaced video;

performing an initial search for each macroblock in high-motion video and quality-demanded video applications, in order to find the most promising search center for the remaining local search, by utilizing either adaptive rood pattern with a fixed rood-arm length adaptively determined or motion adaptive pattern, both based on available neighboring motion vectors from the spatial and/or temporal domains as well as, optionally, global motion vector computed;

conducting local search for each macroblock using a small diamond pattern;

conducting an adaptive threshold pre-judgment for detecting no-motion blocks in the video frames;

computing global motion activity for each video frame;

computing local motion activity for each macroblock;

adapting a search range for each macroblock of each video frame;

Claim 2. The method of claim 1 wherein the profiles as are selected from a group of profile options consisting of simple profile, basic profile, pattern adaptive profile, and main profile, being integrated together to form a scalable block-matching motion estimation architecture;

Claim 3. The method of claim 1 wherein all the search patterns thus designed and presented in the associate figures for non-interlaced and interlaced video; furthermore, the pattern adaptive method could be used for any given pair of two different search patterns for non-interlaced video and for interlaced video, respectively.

Claim 4. The method of claim 1 wherein the pattern adaptive profile is adaptively exploited based on the function of available neighboring motion vectors obtained from the region-of-support of the current macroblock.

Claim 5. The method of claim 1 wherein the adaptive rood pattern adaptively determined for each macroblock consists of a regular rood shape with four equal-

10007223-110201

length rood-arms plus one predicted search position, and an optional global motion vector;

Claim 6. The method of claim 1 wherein the motion adaptive pattern adaptively determined for each macroblock consists of an irregularly formed search pattern based on available motion-vector positions obtained from the region-of-support of the current macroblock;

Claim 7. The method of claim 1 wherein the sub-sampled pixels are achieved by regularly skipping the pixels in both horizontal and vertical directions for reduced computation of macroblock matching distance and thus increased speed-up of the motion-vector search process;

Claim 8. The method of claim 1 wherein the pre-determined application requirements and system's processing modes are: bit rates and delay latency required by selected video applications, network traffic conditions, targeted quality of service, block sizes, selected motion-estimation prediction modes from the group consisting of normal prediction, advanced prediction, frame prediction, field prediction, full-pel, half-pel or quarter-pel prediction, selected video object planes having different priorities and quality requirements.

Claim 9. The method of claim 1 wherein the detection of no-motion activity blocks is based on a pre-judgement threshold map, which is a function of sum of absolute difference plus a deviation factor.

Claim 10. The method of claim 1 wherein the computation of the global motion activity for each video frame comprises:

computing the mean and standard deviation of all of the motion vectors in a video frame, a grade of global motion activity being defined as the maximum of the absolute values of the components of the mean motion vector plus three times the standard deviation.

Claim 11. The method of claim 1 wherein the computation of local motion activity at each block position comprises:

determining city-block lengths of the motion vectors of those blocks that lie in the region-of-support of a current macroblock, a maximum of the city-blocks length used to measure a grade of local motion activity;

classifying local motion activity at the current block position into one of the three classes: "low", "medium" and "high", based on three non-overlapping ranges of the grade of local motion activity.

- Claim 12. The method of claim 1 wherein the profile can be chosen from the profile set based on the grade of local motion activity measured.
- Claim 13. The method of claim 1 further comprising adapting the search range at each block for motion estimation, wherein the search range of a current frame is adaptively determined based on the mean and variance information of a reference frame.
- Claim 14. The method of claim 1 wherein both the large diamond search pattern and small diamond search pattern are individually elongated in both vertical directions by one pixel each for matching interlace video data structure.
- Claim 15. The method of claim 1 further comprises of obtaining an f-code of each video object plane and updating the f-code for video indexing and retrieval.
- Claim 16. A block-matching motion estimation apparatus for use of a 2-D planar systolic array architecture, said apparatus comprising:
- means for simultaneously inputting each row of reference-block data of three checking points in a large diamond and two checking points in a small diamond, into the systolic array from a memory;
  - means for inputting each row of current-block data into the systolic array from a memory;
  - means for computing sum of absolute difference (SAD) with a planar systolic array of 3 columns and P rows, wherein P is the width of the current block;
  - means for comparison of the SAD values for nine checking points in a large diamond and four checking points in a small diamond to identify a final motion vector, where the smallest SAD occurred, and
  - memory architecture for storing search-area data and current-block data.
- Claim 17. The block-matching motion estimation apparatus as claimed in claim 16, wherein said apparatus further comprises:
- a plurality of processing elements arranged as a planar array of 3 columns and P rows, wherein P is a width of a current block;
  - means for interconnection of the processing elements, wherein each processing element receives source data on the current-block from an adjacent processing element in the same row of the array;
  - means for interconnection of the processing elements, wherein each processing element receives reference-block data on the checking points in the search window from an adjacent processing element in a diagonal position of the array, and
  - means for interconnection of the processing elements, wherein each processing element sends an intermediate result of a SAD to an adjacent processing element

in the same column of the array.

Claim 18. The apparatus of claim 16 further comprising a circuit for pipeline inputting each row of current-block data into the systolic array from a memory, wherein said circuit comprises:

a plurality of delay registers for data to be inputted in each column except for the first one, wherein the number of the registers for each column are increased by one, counting from one.

Claim 19. The apparatus of claim 18 further comprising a second circuit for simultaneously inputting each row of reference-block data of three checking points into the systolic array from a memory, wherein said second circuit comprises:

means for connecting a plurality of memory modules to the systolic array, at least two barrel shifters being employed to select correct data for input;  
means for multiplexing by selectively inputting the reference-block data from at least two barrel shifters, and  
a plurality of delay registers for receiving the data to be inputted in each column except the first three, wherein the number of the registers for each column are increased by one, counting from one.

Claim 20. The apparatus of claim 16 wherein the memory means for storing reference data of the checking points are  $P+8$  memory modules for storing reference data, wherein  $P$  is the width of the current block.

Claim 21. The apparatus of claim 17 wherein the processing element is composed of one subtractor, one absolute operator, one adder and three registers, and each intermediate result is outputted after the current and reference data are inputted.

Claim 22. The apparatus of claim 19 wherein each barrel-shifter is composed of  $P+8$  shift registers, wherein  $P$  is the width of the current block; data stored in the registers is shiftable from or to an adjacent neighborhood by a shift operation, one shift completed in one cycle;  
each register in each barrel-shifters being connected to one memory module for data input, and wherein  $P+2$  out of  $P+8$  registers in each barrel-shifter is directly connected to the multiplex means.

Claim 23. The apparatus of claim 19 wherein the multiplex means is comprised of  $P+2$  2-to-1 multiplexers.

Claim 24. The apparatus of claim 20 wherein the memory module stores up to  $P$  pixels of the reference data in a search window, where  $P$  is the width of the current block,

and an address pointer of the memory module can shift forwards and backwards.

Claim 25. A method for using motion vectors to identify moving video objects in a plurality of video frames and the corresponding motion trajectories of the video objects comprising:

using an adaptive nonlinear filtering technique to smooth a motion vectors field;  
clustering to segment motion vector field into homogeneous motion regions that have coherent motion individually; and,  
using bi-directional motion tracking to form video objects and individually track each video object to obtain the motion trajectories thereof.

Claim 26. The method of claim 25 wherein the filtering technique comprises:

classifying each motion vector to be one of four types, true motion vector, isolated irregular motion vector, non-isolated irregular motion vector and edge motion vector;  
selecting the filtering technique for each type identified, the filtering technique consisting of no filtering for a true motion vector, a standard vector median filter for an irregular motion vector and a fuzzy-weighted median filter for an edge motion vector.

Claim 27. The method of claim 26 wherein the filtering is adjusted automatically in respond to the actual percentage of irregular motion vectors contained in the video frames.

Claim 28. The method of claim 26 wherein filtering is carried out only if an estimated percentage of irregularities is more than  $T\%$ , wherein  $T$  optionally being equal to 2.

Claim 29. The method of claim 25 wherein for experimenting each cluster number  $c$  out of a pre-selected range, clustering comprises:

sequentially splitting the largest cluster, in terms of fuzzy hyper volume measurement, in each outer loop iteration into two smaller clusters;  
recursively updating the membership functions in the inner loop and identifying the new cluster centers; and  
computing the cluster validity measurement  $V_c$ .

Claim 30. The method of claim 25 wherein bi-directional motion tracking comprises:

performing a bi-directional projection of video objects from a previous frame and segmented regions from a future frame onto a current frame;  
conducting a validation, and merging and splitting on connected and non-connected motion regions to form valid video objects, and tracking the video objects; and,

performing a Kalman filter smoothing to obtain smoothed motion trajectories for the video objects.

Claim 31. The method of claim 25 wherein the motion vector can be substituted by image pixel intensity or another type of input data, such as optical flow, vector data.

Claim 32. A method for recognizing a handwriting curve or retrieving the most resembled information items from a database of storing such type of items comprising:

performing chain coding for each sample stored in the database and the input sample to be recognized or matched against the database;  
measuring the degree of similarity score between the input chain code and each chain-coded information item stored in database, using evolutionary alignment; and  
recognizing the handwritten curve or identifying the most matched information items from the database.

Claim 33. The method of claim 32 wherein evolutionary alignment for each pair of chain-coded information sequences under matching comprises:

exploiting concave-gap penalty for calculating the cost of opening up a gap for insertion and deletion of a number of codes  
exploiting complex scoring matrix for calculating the cost of replacing one code by another code.

Claim 34. The method of claim 33 wherein the concave gap penalty uses one constant value or the value of a mathematical function for the cost of opening up a gap and one constant value or the value of a mathematical function for the cost of inserting or deleting a number of consecutive codes.

Claim 35. The method of claim 33 wherein the complex scoring matrix uses a constant value or the value of a mathematical function for the cost of replacing one code by another code.

Claim 36. The method of claim 32 wherein the handwritten curves of alphanumeric characters and symbols are pre-stored in database for the user-dependent handwriting application.

Claim 37. The method of claim 32 further comprises resizing the received sample or database samples within the chosen fix-sized bounding box for curve-size normalization to increase recognition or retrieval accuracy.

10007223-110204

Claim 38. The method of claim 32 wherein the information items are applicable to musical notes and audio tones extracted from the musical signal, from which the variations of this information over time are treated as curves and then converted to chain-coded information sequences for matching and retrieval.

10007223-10004